Gene Annotation

Fatemeh

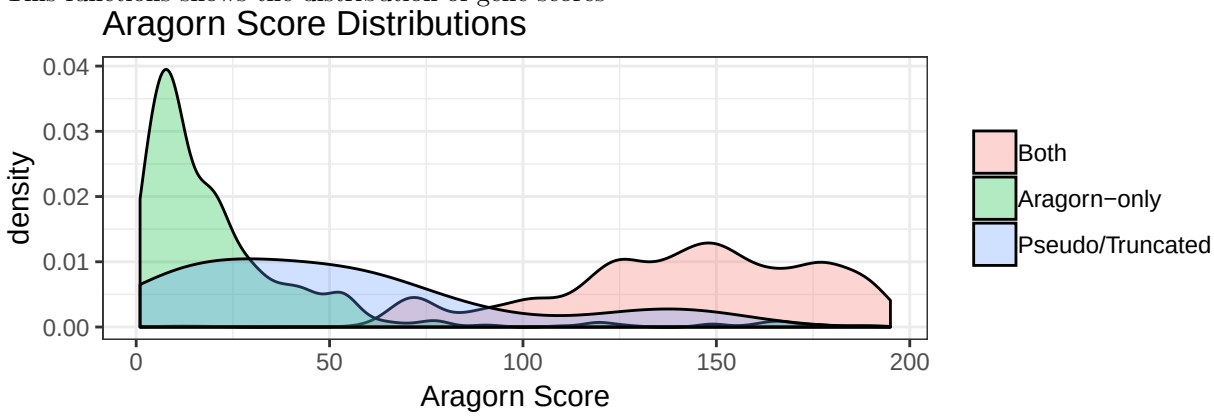
June 17, 2019

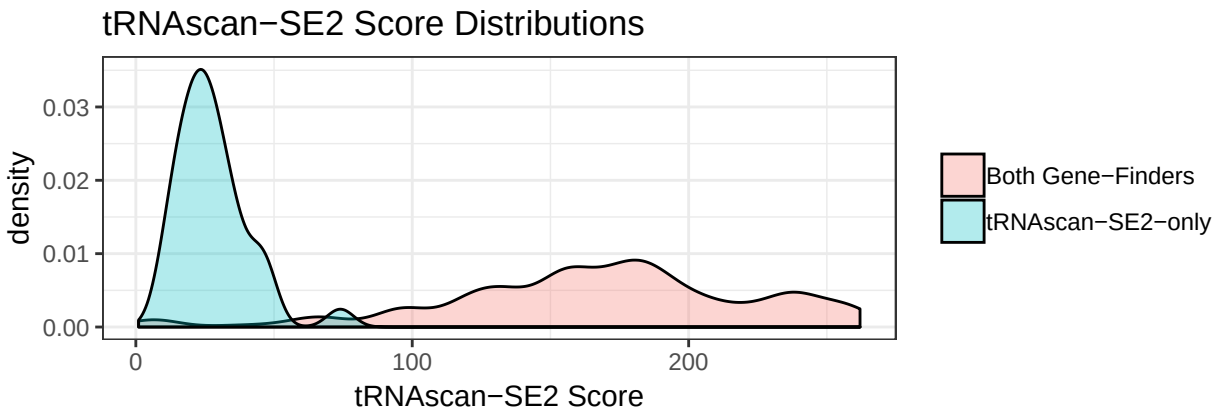
FinalGeneSetAnnotation

This script has the fallowing functions:

1. annotate.final.geneset()
This will take integrated_tse_ara.txt as input, filters genes based on sme criteria and prepares the final genes set.

2. Score.visualization()
This functions shows the distribution of gene scores



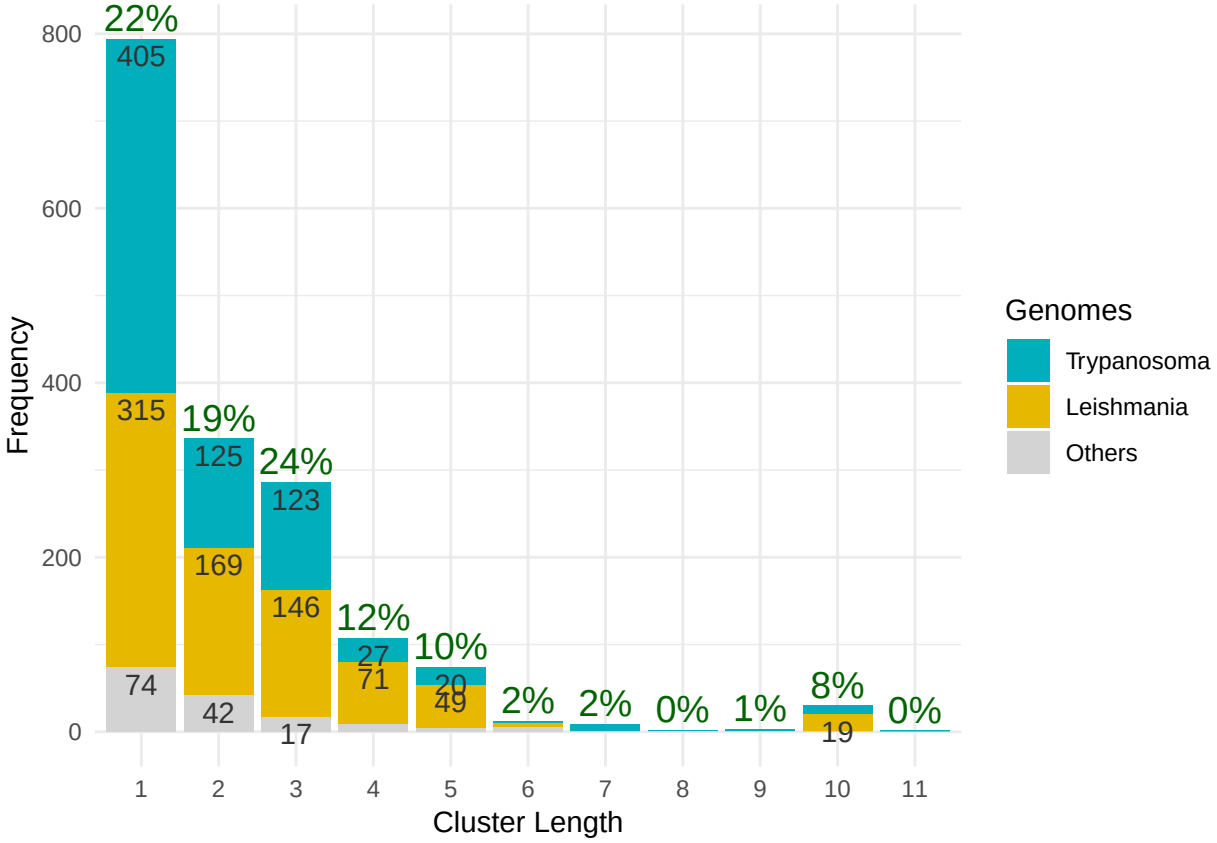


2. create.summary.table()
   This creates a summary table of selected genes.

```
##          Annotation Intersection ARAonly Union
## 1             #tRNA         3554      36  3591
## 2             #N/#G           74      98    75
## 3  Min Gene Length            68      71    68
## 4  Max Gene Length            88     206   206
## 5           %intron            2      28     3
## 6                %G           32      33    32
## 7                %C           26      26    26
## 8                %T           23      23    23
## 9                %A           19      18    19
## 10               A           210       2   212
## 11               C            63       1    64
## 12               D           105       1   106
## 13               E           160       1   161
## 14               F           104       2   106
## 15               G           228       3   231
## 16               H            79       4    83
## 17               I           171       1   172
## 18               K           183       1   184
## 19               L           334       6   340
## 20               M            97       0    97
## 21               N           125       0   125
## 22               P           200       0   200
## 23               Q           161       0   161
## 24               R           348       2   350
## 25               S           227       7   234
## 26               T           218       4   222
## 27               V           235       0   235
## 28               W            52       1    53
## 29               X            76       0    76
## 30               Y            88       0    88
## 31               Z            71       0    71
## 32              ??            19       0    20
```

3. clustersize.dist.visualize()
   This function visualizes the Cluster size distribution for three categories of TryTryp genomes. Labels in green on top of each bar show the percentage of total number of genes as cluster of a specific length. Each color refers to one category of TriTryp genomes. Numbers within each color section of the bar shows the counts of clusters with a specific length.

```
## Scale for 'fill' is already present. Adding another scale for 'fill',
## which will replace the existing scale.
```



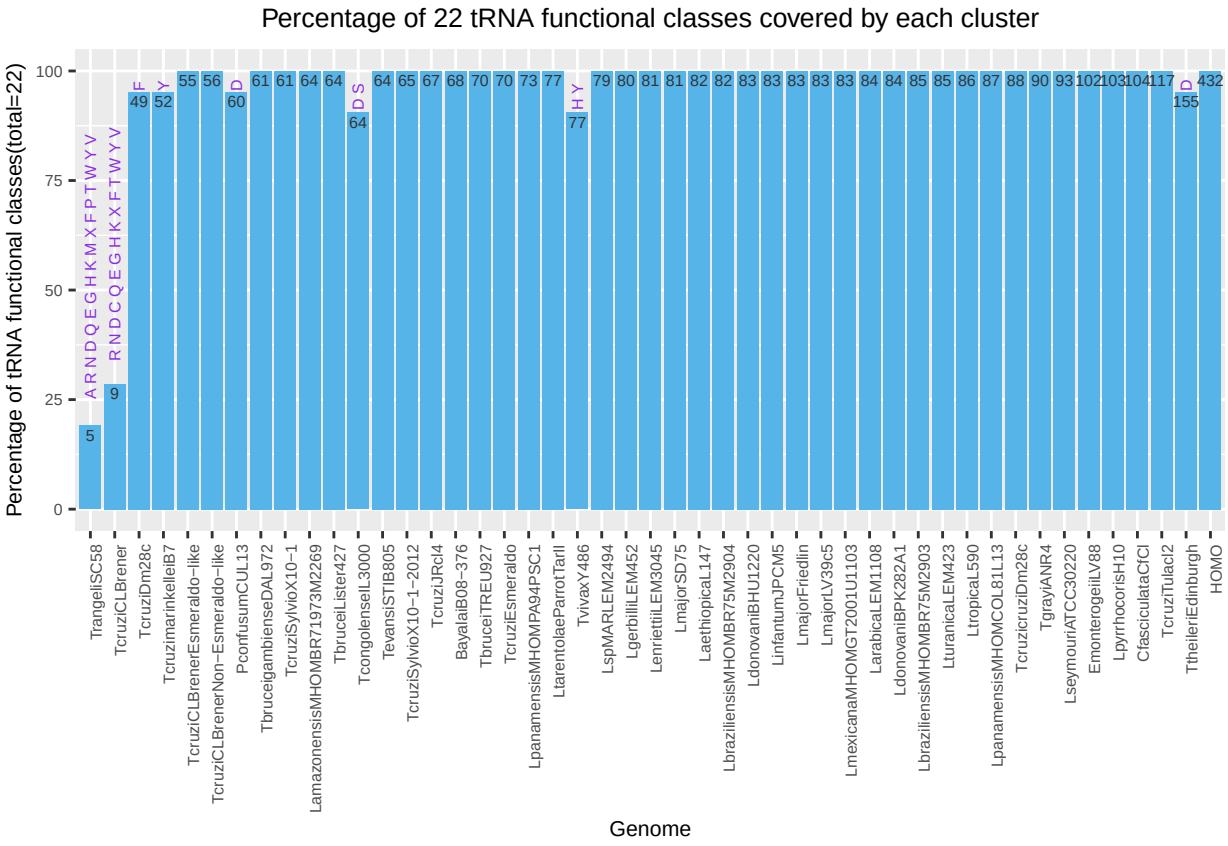
3. prepare.tsfm.input()
Writing the seleted genes in file tsfm_input_geneset.txt.

The gene set tsfm_input_geneset.txt is passed to the script TriTrypAlignment.R to be aligned with Human tRNA genes.

TriTrypAlignment.R

Alignment Steps:
1. Genes from tsfm_input_geneset.txt are read and functional classes are added at the end of the geneID
2. Variable arms are removed based on reported secondary structure from genefinders
3. Gene introns are removed based in the secondary sctructure
4. the result is merged with the Human tRNA genes (the headers for Homo genes are also updated) and the result is saved in file coveainput.fasta.
5. coveainput.fasta is aligned to the Eukaryote model using covea
6. the result (Aligned_TriTryp_Homo.covea) is edited based on the fallowing criteria in order:
a) sites that have more than 98% gap are removed
b) sequences with more than 3 gaps are removed
c) sequences with two or more gaps next to eachother are removed
7. the alignment result is saved as fasta file in Aligned_TriTryp_Homo.fasta, with secondary structure saved as Aligned_TriTryp_Homo_structfile.txt

SplitAlignedGenes.R
The alignment Result Aligned_TriTryp_Homo.fasta will be passed to this script to be splitted either by genome, or clusters of genomes.
The result fasta file for each genome is saved as a file in tsfm/input folder.
The missing functional class for each genome or cluster of genomes is visualized as a bar plot.



Fasta gene files will be splitted based on tRNA functional class by running script splitFuncClass.sh.